Some Practical Tips And Feature Selection

Nipun Batra and teaching staff

IIT Gandhinagar

July 30, 2025

$Feature_1$	$Feature_2$	...	$Feature_d$
True	False	...	False
False	True	...	False
True	True	...	False
⋮	⋮	⋮	
True	True	...	True

$Feature_1$	$Feature_2$	...	$Feature_d$
True	False	...	False
False	True	...	False
True	True	...	False
$\vdots$	$\vdots$	$\vdots$	
True	True	...	True

The entries of the table denote if the feature is used for creating a model. In total we have 2^d models: training models using exhaustive enumeration is very expensive!

Stepwise Forward Selection

$F = \{\}$
for i = 1 to K

$$F_i = \underset{feature \notin F}{\operatorname{argmin}} \text{Loss}(F \cup \text{feature})$$

$$F = F \cup F_i$$

Loss(*features*) denotes the loss incurred by the model trained with *features*.

Stepwise Forward Selection for California Housing Data

Now we will be doing SFS on the California Housing Dataset. We will try to predict the median-selling price(in thousands of dollars) for households in the neighbourhood.

Stepwise Forward Selection for California Housing Data

Iteration	Added Feature	MSE
1	Median Income of block	0.97
2	Avg. number of rooms in the block	0.63
3	Latitude	0.65
4	Longitude	0.66

This shows except the first two features, everything else are unimportant features.

Stepwise Backward Selection

Same as SFS, but in opposite direction
Remove feature, which reduces the accuracy the least(uninmportant).

Time Complexity Analysis

Both SFS and SBS are $O(d^2)$ algorithms, where d is the number of features.

$$\implies (d) + (d-1) + (d-2) + \cdots + (1)$$

$$\implies \frac{d(d-1)}{2}$$

$$\implies d(d-1) \implies d^2$$